


LEARNING MADE EASY



2nd Edition

Python® for Data Science

**for
dummies®**
A Wiley Brand



Learn Python data analysis
programming and statistics

Write code in the
cloud with Google Colab™

Wrangle data and
visualize information

John Paul Mueller
Luca Massaron

*Authors of Machine Learning for Dummies
and Artificial Intelligence For Dummies*

Python® for Data Science

**for
dummies®**
A Wiley Brand



Python[®] for Data Science

2nd Edition

**by John Paul Mueller
and Luca Massaron**

**for
dummies[®]**
A Wiley Brand

Python® for Data Science For Dummies®, 2nd Edition

Published by: **John Wiley & Sons, Inc.**, 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Copyright © 2019 by John Wiley & Sons, Inc., Hoboken, New Jersey

Media and software compilation copyright © 2019 by John Wiley & Sons, Inc. All rights reserved.

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, For Dummies, the Dummies Man logo, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and may not be used without written permission. Python is a registered trademark of Python Software Foundation Corporation. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit <https://hub.wiley.com/community/support/dummies>.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2018967877

ISBN: 978-1-119-54762-4; ISBN: 978-1-119-54766-2 (ebk); ISBN: 978-1-119-54764-8 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Contents at a Glance

Introduction	1
Part 1: Getting Started with Data Science and Python	7
CHAPTER 1: Discovering the Match between Data Science and Python	9
CHAPTER 2: Introducing Python's Capabilities and Wonders	21
CHAPTER 3: Setting Up Python for Data Science	39
CHAPTER 4: Working with Google Colab	59
Part 2: Getting Your Hands Dirty with Data	81
CHAPTER 5: Understanding the Tools	83
CHAPTER 6: Working with Real Data	99
CHAPTER 7: Conditioning Your Data	121
CHAPTER 8: Shaping Data	149
CHAPTER 9: Putting What You Know in Action	169
Part 3: Visualizing Information	183
CHAPTER 10: Getting a Crash Course in Matplotlib	185
CHAPTER 11: Visualizing the Data	201
Part 4: Wrangling Data	227
CHAPTER 12: Stretching Python's Capabilities	229
CHAPTER 13: Exploring Data Analysis	251
CHAPTER 14: Reducing Dimensionality	275
CHAPTER 15: Clustering	295
CHAPTER 16: Detecting Outliers in Data	313
Part 5: Learning from Data	327
CHAPTER 17: Exploring Four Simple and Effective Algorithms	329
CHAPTER 18: Performing Cross-Validation, Selection, and Optimization	347
CHAPTER 19: Increasing Complexity with Linear and Nonlinear Tricks	371
CHAPTER 20: Understanding the Power of the Many	411
Part 6: The Part of Tens	429
CHAPTER 21: Ten Essential Data Resources	431
CHAPTER 22: Ten Data Challenges You Should Take	437
Index	447

Table of Contents

INTRODUCTION	1
About This Book	1
Foolish Assumptions	3
Icons Used in This Book	4
Beyond the Book	4
Where to Go from Here	5
 PART 1: GETTING STARTED WITH DATA SCIENCE AND PYTHON	 7
CHAPTER 1: Discovering the Match between Data Science and Python	9
Defining the Sexiest Job of the 21st Century	11
Considering the emergence of data science	12
Outlining the core competencies of a data scientist	12
Linking data science, big data, and AI	13
Understanding the role of programming	14
Creating the Data Science Pipeline	14
Preparing the data	15
Performing exploratory data analysis	15
Learning from data	15
Visualizing	15
Obtaining insights and data products	16
Understanding Python's Role in Data Science	16
Considering the shifting profile of data scientists	16
Working with a multipurpose, simple, and efficient language	17
Learning to Use Python Fast	18
Loading data	19
Training a model	19
Viewing a result	19
 CHAPTER 2: Introducing Python's Capabilities and Wonders	 21
Why Python?	22
Grasping Python's Core Philosophy	23
Contributing to data science	23
Discovering present and future development goals	24

Working with Python.	25
Getting a taste of the language.	25
Understanding the need for indentation.	26
Working at the command line or in the IDE	27
Performing Rapid Prototyping and Experimentation	31
Considering Speed of Execution	32
Visualizing Power.	33
Using the Python Ecosystem for Data Science	35
Accessing scientific tools using SciPy	35
Performing fundamental scientific computing using NumPy.	36
Performing data analysis using pandas.	36
Implementing machine learning using Scikit-learn	36
Going for deep learning with Keras and TensorFlow.	37
Plotting the data using matplotlib.	38
Creating graphs with NetworkX.	38
Parsing HTML documents using Beautiful Soup.	38
CHAPTER 3: Setting Up Python for Data Science	39
Considering the Off-the-Shelf Cross-Platform	
Scientific Distributions	40
Getting Continuum Analytics Anaconda	40
Getting Enthought Deployment Manager	41
Getting WinPython	42
Installing Anaconda on Windows	42
Installing Anaconda on Linux.	46
Installing Anaconda on Mac OS X	47
Downloading the Datasets and Example Code.	48
Using Jupyter Notebook	49
Defining the code repository.	50
Understanding the datasets used in this book.	57
CHAPTER 4: Working with Google Colab	59
Defining Google Colab	60
Understanding what Google Colab does.	60
Considering the online coding difference	61
Using local runtime support	63
Getting a Google Account.	63
Creating the account.	64
Signing in	64
Working with Notebooks	65
Creating a new notebook.	65
Opening existing notebooks	66
Saving notebooks	68
Downloading notebooks	71

Performing Common Tasks	71
Creating code cells	71
Creating text cells	72
Creating special cells	73
Editing cells	74
Moving cells	75
Using Hardware Acceleration	75
Executing the Code	76
Viewing Your Notebook	76
Displaying the table of contents	77
Getting notebook information	77
Checking code execution	78
Sharing Your Notebook	79
Getting Help	80

PART 2: GETTING YOUR HANDS DIRTY WITH DATA 81

CHAPTER 5: Understanding the Tools	83
Using the Jupyter Console	84
Interacting with screen text	84
Changing the window appearance	86
Getting Python help	87
Getting IPython help	89
Using magic functions	90
Discovering objects	91
Using Jupyter Notebook	93
Working with styles	93
Restarting the kernel	94
Restoring a checkpoint	95
Performing Multimedia and Graphic Integration	96
Embedding plots and other images	96
Loading examples from online sites	96
Obtaining online graphics and multimedia	96
CHAPTER 6: Working with Real Data	99
Uploading, Streaming, and Sampling Data	100
Uploading small amounts of data into memory	101
Streaming large amounts of data into memory	102
Generating variations on image data	103
Sampling data in different ways	104
Accessing Data in Structured Flat-File Form	105
Reading from a text file	106
Reading CSV delimited format	107
Reading Excel and other Microsoft Office files	109

	Sending Data in Unstructured File Form	111
	Managing Data from Relational Databases.....	113
	Interacting with Data from NoSQL Databases	115
	Accessing Data from the Web	116
CHAPTER 7:	Conditioning Your Data	121
	Juggling between NumPy and pandas	122
	Knowing when to use NumPy	122
	Knowing when to use pandas	122
	Validating Your Data	124
	Figuring out what's in your data	124
	Removing duplicates.....	126
	Creating a data map and data plan	126
	Manipulating Categorical Variables	129
	Creating categorical variables	130
	Renaming levels.....	131
	Combining levels	132
	Dealing with Dates in Your Data	133
	Formatting date and time values	134
	Using the right time transformation.....	135
	Dealing with Missing Data	136
	Finding the missing data	136
	Encoding missingness.....	137
	Imputing missing data	138
	Slicing and Dicing: Filtering and Selecting Data	139
	Slicing rows.....	140
	Slicing columns	140
	Dicing.....	141
	Concatenating and Transforming	142
	Adding new cases and variables	142
	Removing data	144
	Sorting and shuffling.....	145
	Aggregating Data at Any Level.....	146
CHAPTER 8:	Shaping Data	149
	Working with HTML Pages	150
	Parsing XML and HTML	150
	Using XPath for data extraction	151
	Working with Raw Text	153
	Dealing with Unicode	153
	Stemming and removing stop words	153
	Introducing regular expressions.....	155
	Using the Bag of Words Model and Beyond	158
	Understanding the bag of words model	159

Working with n-grams.	161
Implementing TF-IDF transformations.	162
Working with Graph Data.	165
Understanding the adjacency matrix.	165
Using NetworkX basics.	166
CHAPTER 9: Putting What You Know in Action	169
Contextualizing Problems and Data.	170
Evaluating a data science problem.	171
Researching solutions.	173
Formulating a hypothesis.	174
Preparing your data.	175
Considering the Art of Feature Creation.	175
Defining feature creation.	175
Combining variables.	176
Understanding binning and discretization.	177
Using indicator variables.	177
Transforming distributions.	178
Performing Operations on Arrays.	178
Using vectorization.	179
Performing simple arithmetic on vectors and matrices.	179
Performing matrix vector multiplication.	180
Performing matrix multiplication.	181
PART 3: VISUALIZING INFORMATION	183
CHAPTER 10: Getting a Crash Course in Matplotlib.	185
Starting with a Graph.	186
Defining the plot.	186
Drawing multiple lines and plots.	187
Saving your work to disk.	188
Setting the Axis, Ticks, Grids.	189
Getting the axes.	189
Formatting the axes.	190
Adding grids.	191
Defining the Line Appearance.	192
Working with line styles.	193
Using colors.	194
Adding markers.	195
Using Labels, Annotations, and Legends.	197
Adding labels.	198
Annotating the chart.	198
Creating a legend.	199

CHAPTER 11: Visualizing the Data	201
Choosing the Right Graph	202
Showing parts of a whole with pie charts	202
Creating comparisons with bar charts	203
Showing distributions using histograms	205
Depicting groups using boxplots	206
Seeing data patterns using scatterplots	208
Creating Advanced Scatterplots	209
Depicting groups	209
Showing correlations	211
Plotting Time Series	212
Representing time on axes	212
Plotting trends over time	214
Plotting Geographical Data	216
Using an environment in Notebook	217
Getting the Basemap toolkit	218
Dealing with deprecated library issues	218
Using Basemap to plot geographic data	220
Visualizing Graphs	221
Developing undirected graphs	222
Developing directed graphs	224
 PART 4: WRANGLING DATA	 227
 CHAPTER 12: Stretching Python's Capabilities	 229
Playing with Scikit-learn	230
Understanding classes in Scikit-learn	230
Defining applications for data science	231
Performing the Hashing Trick	234
Using hash functions	235
Demonstrating the hashing trick	235
Working with deterministic selection	239
Considering Timing and Performance	240
Benchmarking with timeit	241
Working with the memory profiler	244
Running in Parallel on Multiple Cores	247
Performing multicore parallelism	248
Demonstrating multiprocessing	248
 CHAPTER 13: Exploring Data Analysis	 251
The EDA Approach	252
Defining Descriptive Statistics for Numeric Data	253
Measuring central tendency	254
Measuring variance and range	255

Working with percentiles	256
Defining measures of normality	257
Counting for Categorical Data	259
Understanding frequencies	259
Creating contingency tables	261
Creating Applied Visualization for EDA	261
Inspecting boxplots	262
Performing t-tests after boxplots	263
Observing parallel coordinates	264
Graphing distributions	265
Plotting scatterplots	266
Understanding Correlation	268
Using covariance and correlation	268
Using nonparametric correlation	270
Considering the chi-square test for tables	271
Modifying Data Distributions	272
Using different statistical distributions	272
Creating a Z-score standardization	273
Transforming other notable distributions	273
CHAPTER 14: Reducing Dimensionality	275
Understanding SVD	276
Looking for dimensionality reduction	277
Using SVD to measure the invisible	279
Performing Factor Analysis and PCA	280
Considering the psychometric model	280
Looking for hidden factors	281
Using components, not factors	282
Achieving dimensionality reduction	282
Squeezing information with t-SNE	283
Understanding Some Applications	285
Recognizing faces with PCA	285
Extracting topics with NMF	289
Recommending movies	291
CHAPTER 15: Clustering	295
Clustering with K-means	297
Understanding centroid-based algorithms	298
Creating an example with image data	299
Looking for optimal solutions	301
Clustering big data	304
Performing Hierarchical Clustering	305
Using a hierarchical cluster solution	307
Using a two-phase clustering solution	308
Discovering New Groups with DBScan	310

CHAPTER 16: Detecting Outliers in Data	313
Considering Outlier Detection	314
Finding more things that can go wrong	315
Understanding anomalies and novel data	316
Examining a Simple Univariate Method	317
Leveraging on the Gaussian distribution	319
Making assumptions and checking out	320
Developing a Multivariate Approach	322
Using principal component analysis	322
Using cluster analysis for spotting outliers	324
Automating detection with Isolation Forests	325
 PART 5: LEARNING FROM DATA	 327
 CHAPTER 17: Exploring Four Simple and Effective Algorithms	 329
Guessing the Number: Linear Regression	329
Defining the family of linear models	330
Using more variables	331
Understanding limitations and problems	333
Moving to Logistic Regression	334
Applying logistic regression	335
Considering when classes are more	336
Making Things as Simple as Naïve Bayes	337
Finding out that Naïve Bayes isn't so naïve	339
Predicting text classifications	340
Learning Lazily with Nearest Neighbors	342
Predicting after observing neighbors	343
Choosing your k parameter wisely	344
 CHAPTER 18: Performing Cross-Validation, Selection, and Optimization	 347
Pondering the Problem of Fitting a Model	348
Understanding bias and variance	349
Defining a strategy for picking models	350
Dividing between training and test sets	354
Cross-Validating	356
Using cross-validation on k folds	357
Sampling stratifications for complex data	358
Selecting Variables Like a Pro	360
Selecting by univariate measures	360
Using a greedy search	362
Pumping Up Your Hyperparameters	363
Implementing a grid search	364
Trying a randomized search	368

CHAPTER 19: Increasing Complexity with Linear and Nonlinear Tricks	371
Using Nonlinear Transformations	372
Doing variable transformations	372
Creating interactions between variables	375
Regularizing Linear Models	379
Relying on Ridge regression (L2)	380
Using the Lasso (L1)	381
Leveraging regularization	382
Combining L1 & L2: Elasticnet	382
Fighting with Big Data Chunk by Chunk	383
Determining when there is too much data	383
Implementing Stochastic Gradient Descent	383
Understanding Support Vector Machines	387
Relying on a computational method	387
Fixing many new parameters	390
Classifying with SVC	392
Going nonlinear is easy	398
Performing regression with SVR	399
Creating a stochastic solution with SVM	401
Playing with Neural Networks	406
Understanding neural networks	407
Classifying and regressing with neurons	408
CHAPTER 20: Understanding the Power of the Many	411
Starting with a Plain Decision Tree	412
Understanding a decision tree	412
Creating trees for different purposes	415
Making Machine Learning Accessible	418
Working with a Random Forest classifier	420
Working with a Random Forest regressor	421
Optimizing a Random Forest	422
Boosting Predictions	424
Knowing that many weak predictors win	424
Setting a gradient boosting classifier	425
Running a gradient boosting regressor	426
Using GBM hyperparameters	427
PART 6: THE PART OF TENS	429
CHAPTER 21: Ten Essential Data Resources	431
Discovering the News with Subreddit	432
Getting a Good Start with KDnuggets	432
Locating Free Learning Resources with Quora	432

Gaining Insights with Oracle's Data Science Blog	433
Accessing the Huge List of Resources on Data Science Central	433
Learning New Tricks from the Aspirational Data Scientist	434
Obtaining the Most Authoritative Sources at Udacity	435
Receiving Help with Advanced Topics at Conductrics	435
Obtaining the Facts of Open Source Data Science from Masters . .	436
Zeroing In on Developer Resources with Jonathan Bower	436
CHAPTER 22: Ten Data Challenges You Should Take	437
Meeting the Data Science London + Scikit-learn Challenge	438
Predicting Survival on the Titanic	438
Finding a Kaggle Competition that Suits Your Needs.	439
Honing Your Overfit Strategies	440
Trudging Through the MovieLens Dataset	440
Getting Rid of Spam E-mails.	441
Working with Handwritten Information.	442
Working with Pictures.	443
Analyzing Amazon.com Reviews.	444
Interacting with a Huge Graph	444
INDEX	447

Introduction

Data is increasingly used for every possible purpose, and many of those purposes elude attention, but every time you get on the Internet, you generate even more. It's not just you, either; the growth of the Internet has been phenomenal, according to Internet World Stats (<https://www.internetworldstats.com/emarketing.htm>). Data science turns this huge amount of data into something useful — something that you use absolutely every day to perform an amazing array of tasks or to obtain services from someone else.

In fact, you've probably used data science in ways that you never expected. For example, when you used your favorite search engine this morning to look for something, it made suggestions on alternative search terms. Those terms are supplied by data science. When you went to the doctor last week and discovered the lump you found wasn't cancer, the doctor likely made her prognosis with the help of data science. In fact, you might work with data science every day and not even know it. *Python for Data Science For Dummies*, 2nd Edition not only gets you started using data science to perform a wealth of practical tasks but also helps you realize just how many places data science is used. By knowing how to answer data science problems and where to employ data science, you gain a significant advantage over everyone else, increasing your chances at promotion or that new job you really want.

About This Book

The main purpose of *Python for Data Science For Dummies*, 2nd Edition is to take the scare factor out of data science by showing you that data science is not only really interesting but also quite doable using Python. You might assume that you need to be a computer science genius to perform the complex tasks normally associated with data science, but that's far from the truth. Python comes with a host of useful libraries that do all the heavy lifting for you in the background. You don't even realize how much is going on, and you don't need to care. All you really need to know is that you want to perform specific tasks, and Python makes these tasks quite accessible.

Part of the emphasis of this book is on using the right tools. You start with Anaconda, a product that includes IPython and Jupyter Notebook — two tools that take the sting out of working with Python. You experiment with IPython in a fully interactive environment. The code you place in Jupyter Notebook (also called just Notebook throughout the book) is presentation quality, and you can mix a number of presentation elements right there in your document. It's not really like using a development environment at all. To make this book easier to use on alternative platforms, you also discover an online Interactive Development Environment application (IDE) named Google Colab that allows you to interact with most, but not quite all, of the book examples using your favorite tablet or (assuming that you can squint well enough) your smart phone.

You also discover some interesting techniques in this book. For example, you can create plots of all your data science experiments using Matplotlib, and this book gives you all the details for doing that. This book also spends considerable time showing you available resources (such as packages) and how you can use Scikit-learn to perform some really interesting calculations. Many people would like to know how to perform handwriting recognition, and if you're one of them, you can use this book to get a leg up on the process.

Of course, you might still be worried about the whole programming environment issue, and this book doesn't leave you in the dark there, either. At the beginning, you find complete installation instructions for Anaconda, which are followed by the methods you need to get started with data science using Jupyter Notebook or Google Colab. The emphasis is on getting you up and running as quickly as possible, and to make examples straightforward and simple so that the code doesn't become a stumbling block to learning.

This second edition of the book provides you with updated examples using Python 3.x so that you're using the most modern version of Python while reading. In addition, you find a stronger emphasis on making examples simpler, but also making the environment more inclusive by adding material on deep learning. Consequently, you get a lot more out of this edition of the book as a result of the input provided by hundreds of readers before you.

To make absorbing the concepts even easier, this book uses the following conventions:

- » Text that you're meant to type just as it appears in the book is in **bold**. The exception is when you're working through a step list: Because each step is bold, the text to type is not bold.
- » When you see words in *italics* as part of a typing sequence, you need to replace that value with something that works for you. For example, if you

see “Type **Your Name** and press Enter,” you need to replace *Your Name* with your actual name.

- » Web addresses and programming code appear in monofont. If you’re reading a digital version of this book on a device connected to the Internet, note that you can click the web address to visit that website, like this: `http://www.dummies.com`.
- » When you need to type command sequences, you see them separated by a special arrow, like this: File ⇨ New File. In this example, you go to the File menu first and then select the New File entry on that menu.

Foolish Assumptions

You might find it difficult to believe that we’ve assumed anything about you — after all, we haven’t even met you yet! Although most assumptions are indeed foolish, we made these assumptions to provide a starting point for the book.

You need to be familiar with the platform you want to use because the book doesn’t offer any guidance in this regard. (Chapter 3 does, however, provide Anaconda installation instructions, and Chapter 4 gets you started with Google Colab.) To provide you with maximum information about Python concerning how it applies to data science, this book doesn’t discuss any platform-specific issues. You really do need to know how to install applications, use applications, and generally work with your chosen platform before you begin working with this book.

You must know how to work with Python. This edition of the book no longer contains a Python primer because you can find such a wealth of tutorials online (see <https://www.w3schools.com/python/> and <https://www.tutorialspoint.com/python/> as examples).

This book isn’t a math primer. Yes, you see lots of examples of complex math, but the emphasis is on helping you use Python and data science to perform analysis tasks rather than teaching math theory. Chapters 1 and 2 give you a better understanding of precisely what you need to know to use this book successfully.

This book also assumes that you can access items on the Internet. Sprinkled throughout are numerous references to online material that will enhance your learning experience. However, these added sources are useful only if you actually find and use them.

Icons Used in This Book

As you read this book, you see icons in the margins that indicate material of interest (or not, as the case may be). This section briefly describes each icon in this book.



TIP

Tips are nice because they help you save time or perform some task without a lot of extra work. The tips in this book are time-saving techniques or pointers to resources that you should try in order to get the maximum benefit from Python or in performing data science–related tasks.



WARNING

We don't want to sound like angry parents or some kind of maniacs, but you should avoid doing anything that's marked with a Warning icon. Otherwise, you might find that your application fails to work as expected, you get incorrect answers from seemingly bulletproof equations, or (in the worst-case scenario) you lose data.



TECHNICAL
STUFF

Whenever you see this icon, think advanced tip or technique. You might find these tidbits of useful information just too boring for words, or they could contain the solution you need to get a program running. Skip these bits of information whenever you like.



REMEMBER

If you don't get anything else out of a particular chapter or section, remember the material marked by this icon. This text usually contains an essential process or a bit of information that you must know to work with Python or to perform data science–related tasks successfully.

Beyond the Book

This book isn't the end of your Python or data science experience — it's really just the beginning. We provide online content to make this book more flexible and better able to meet your needs. That way, as we receive e-mail from you, we can address questions and tell you how updates to either Python or its associated add-ons affect book content. In fact, you gain access to all these cool additions:

» **Cheat sheet:** You remember using crib notes in school to make a better mark on a test, don't you? You do? Well, a cheat sheet is sort of like that. It provides you with some special notes about tasks that you can do with Python, IPython, IPython Notebook, and data science that not every other person knows. You can find the cheat sheet by going to www.dummies.com, searching this book's title, and scrolling down the page that appears. The cheat sheet contains really

neat information such as the most common programming mistakes that cause people woe when using Python.

- » **Updates:** Sometimes changes happen. For example, we might not have seen an upcoming change when we looked into our crystal ball during the writing of this book. In the past, this possibility simply meant that the book became outdated and less useful, but you can now find updates to the book by searching this book's title at www.dummies.com.

In addition to these updates, check out the blog posts with answers to reader questions and demonstrations of useful book-related techniques at <http://blog.johnmuelเลอร์books.com/>.

- » **Companion files:** Hey! Who really wants to type all the code in the book and reconstruct all those plots manually? Most readers would prefer to spend their time actually working with Python, performing data science tasks, and seeing the interesting things they can do, rather than typing. Fortunately for you, the examples used in the book are available for download, so all you need to do is read the book to learn Python for data science usage techniques. You can find these files at www.dummies.com. Search this book's title, and on the page that appears, scroll down to the image of the book cover and click it. Then click the More about This Book button and on the page that opens, go to the Downloads tab.

Where to Go from Here

It's time to start your Python for data science adventure! If you're completely new to Python and its use for data science tasks, you should start with Chapter 1 and progress through the book at a pace that allows you to absorb as much of the material as possible.

If you're a novice who's in an absolute rush to get going with Python for data science as quickly as possible, you can skip to Chapter 3 with the understanding that you may find some topics a bit confusing later. Skipping to Chapter 5 is okay if you already have Anaconda (the programming product used in the book) installed, but be sure to at least skim Chapter 3 so that you know what assumptions we made when writing this book. If you plan to use your tablet to work with this book, be certain to review Chapter 4 so that you understand the limitations presented by Google Colab in running the example code; not all of the examples work in this IDE. Make sure to install Anaconda with Python version 3.6.5 installed to obtain the best results from the book's source code.

Readers who have some exposure to Python and have Anaconda installed can save reading time by moving directly to Chapter 5. You can always go back to earlier chapters as necessary when you have questions. However, you should understand how each technique works before moving to the next one. Every technique, coding example, and procedure has important lessons for you, and you could miss vital content if you start skipping too much information.

1

Getting Started with Data Science and Python

IN THIS PART . . .

Understanding how Python can make data science easier.

Defining the Python features commonly used for data science.

Creating a Python setup of your own.

Working with Google Colab on alternative devices.

- » Discovering the wonders for data science
- » Exploring how data science works
- » Creating the connection between Python and data science
- » Getting started with Python

Chapter 1

Discovering the Match between Data Science and Python

Data science may seem like one of those technologies that you'd never use, but you'd be wrong. Yes, data science involves the use of advanced math techniques, statistics, and big data. However, data science also involves helping you make smart decisions, creating suggestions for options based on previous choices, and making robots see objects. In fact, people use data science in so many different ways that you literally can't look anywhere or do anything without feeling the effects of data science on your life. In short, data science is the person behind the partition in the experience of the wonderment of technology. Without data science, much of what you accept as typical and expected today wouldn't even be possible. This is the reason that being a data scientist is the sexiest job of the twenty-first century.



REMEMBER

To make data science doable by someone who's less than a math genius, you need tools. You could use any of a number of tools to perform data science tasks, but Python is uniquely suited to making it easier to work with data science. For one thing, Python provides an incredible number of math-related libraries that help you perform tasks with a less-than-perfect understanding of precisely what is

going on. However, Python goes further by supporting multiple coding styles (programming paradigms) and doing other things to make your job easier. Therefore, yes, you could use other languages to write data science applications, but Python reduces your workload, so it's a natural choice for those who really don't want to work hard, but rather to work smart.

This chapter gets you started with Python. Even though this book isn't designed to provide you with a complete Python tutorial, exploring some basic Python issues will reduce the time needed for you to get up to speed. (If you do need a good starting tutorial, please get *Beginning Programming with Python For Dummies*, 2nd Edition, by John Mueller (Wiley). You'll find that the book provides pointers to tutorials and other aids as needed to fill in any gaps that you may have in your Python education.

CHOOSING A DATA SCIENCE LANGUAGE

There are many different programming languages in the world — and most were designed to perform tasks in a certain way or even make it easier for a particular profession's work to be done with greater ease. Choosing the correct tool makes your life easier. It's akin to using a hammer to drive a screw rather than a screwdriver. Yes, the hammer works, but the screwdriver is much easier to use and definitely does a better job. Data scientists usually use only a few languages because they make working with data easier. With this in mind, here are the top languages for data science work in order of preference:

- **Python (general purpose):** Many data scientists prefer to use Python because it provides a wealth of libraries, such as NumPy, SciPy, Matplotlib, pandas, and Scikit-learn, to make data science tasks significantly easier. Python is also a precise language that makes it easy to use multi-processing on large datasets — reducing the time required to analyze them. The data science community has also stepped up with specialized IDEs, such as Anaconda, that implement the Jupyter Notebook concept, which makes working with data science calculations significantly easier (Chapter 3 demonstrates how to use Jupyter Notebook, so don't worry about it in this chapter). Besides all of these things in Python's favor, it's also an excellent language for creating glue code with languages such as C/C++ and Fortran. The Python documentation actually shows how to create the required extensions. Most Python users rely on the language to see patterns, such as allowing a robot to see a group of pixels as an object. It also sees use for all sorts of scientific tasks.
- **R (special purpose statistical):** In many respects, Python and R share the same sorts of functionality but implement it in different ways. Depending on which source you view, Python and R have about the same number of proponents, and some people use Python and R interchangeably (or sometimes in tandem). Unlike

Python, R provides its own environment, so you don't need a third-party product such as Anaconda. However, R doesn't appear to mix with other languages with the ease that Python provides.

- **SQL (database management):** The most important thing to remember about Structured Query Language (SQL) is that it focuses on data rather than tasks. Businesses can't operate without good data management — the data is the business. Large organizations use some sort of relational database, which is normally accessible with SQL, to store their data. Most Database Management System (DBMS) products rely on SQL as their main language, and DBMS usually has a large number of data analysis and other data science features built in. Because you're accessing the data natively, there is often a significant speed gain in performing data science tasks this way. Database Administrators (DBAs) generally use SQL to manage or manipulate the data rather than necessarily perform detailed analysis of it. However, the data scientist can also use SQL for various data science tasks and make the resulting scripts available to the DBAs for their needs.
- **Java (general purpose):** Some data scientists perform other kinds of programming that require a general purpose, widely adapted and popular, language. In addition to providing access to a large number of libraries (most of which aren't actually all that useful for data science, but do work for other needs), Java supports object orientation better than any of the other languages in this list. In addition, it's strongly typed and tends to run quite quickly. Consequently, some people prefer it for finalized code. Java isn't a good choice for experimentation or ad hoc queries.
- **Scala (general purpose):** Because Scala uses the Java Virtual Machine (JVM) it does have some of the advantages and disadvantages of Java. However, like Python, Scala provides strong support for the functional programming paradigm, which uses lambda calculus as its basis (see *Functional Programming For Dummies*, by John Mueller [Wiley] for details). In addition, Apache Spark is written in Scala, which means that you have good support for cluster computing when using this language; — think huge dataset support. Some of the pitfalls of using Scala are that it's hard to set up correctly, it has a steep learning curve, and it lacks a comprehensive set of data science specific libraries.

Defining the Sexiest Job of the 21st Century

At one point, the world viewed anyone working with statistics as a sort of accountant or perhaps a mad scientist. Many people consider statistics and analysis of data boring. However, data science is one of those occupations in which the more you learn, the more you want to learn. Answering one question often spawns more questions that are even more interesting than the one you just answered.

However, the thing that makes data science so sexy is that you see it everywhere and used in an almost infinite number of ways. The following sections provide you with more details on why data science is such an amazing field of study.

Considering the emergence of data science

Data science is a relatively new term. William S. Cleveland coined the term in 2001 as part of a paper entitled “Data Science: An Action Plan for Expanding the Technical Areas of the Field of Statistics.” It wasn’t until a year later that the International Council for Science actually recognized data science and created a committee for it. Columbia University got into the act in 2003 by beginning publication of the *Journal of Data Science*.



REMEMBER

However, the mathematical basis behind data science is centuries old because data science is essentially a method of viewing and analyzing statistics and probability. The first essential use of statistics as a term comes in 1749, but statistics are certainly much older than that. People have used statistics to recognize patterns for thousands of years. For example, the historian Thucydides (in his *History of the Peloponnesian War*) describes how the Athenians calculated the height of the wall of Platea in fifth century BC by counting bricks in an unplastered section of the wall. Because the count needed to be accurate, the Athenians took the average of the count by several soldiers.

The process of quantifying and understanding statistics is relatively new, but the science itself is quite old. An early attempt to begin documenting the importance of statistics appears in the ninth century when Al-Kindi wrote *Manuscript on Deciphering Cryptographic Messages*. In this paper, Al-Kindi describes how to use a combination of statistics and frequency analysis to decipher encrypted messages. Even in the beginning, statistics saw use in practical application of science to tasks that seemed virtually impossible to complete. Data science continues this process, and to some people it might actually seem like magic.

Outlining the core competencies of a data scientist

As is true of anyone performing most complex trades today, the data scientist requires knowledge of a broad range of skills to perform the required tasks. In fact, so many different skills are required that data scientists often work in teams. Someone who is good at gathering data might team up with an analyst and someone gifted in presenting information. It would be hard to find a single person with all the required skills. With this in mind, the following list describes areas in which a data scientist could excel (with more competencies being better):